

Penyarian Data Pelan Akui Melalui Pengimbasan

Taher Buyong

Bong Thai Kwong

Jabatan Geoinformatik

Universiti Teknologi Malaysia

Karung Berkunci 791, 80990 Johor Bahru

Abstrak

Data kadaster merupakan antara data terpenting dalam GIS. Penukaran data kadaster ke dalam bentuk digital dari bentuk salinan keras menggunakan papan kekunci. Kaedah ini ternyata memakan masa dan kos yang tinggi serta terdedah kepada kesilapan manusia. Kaedah pendigitan pula, walaupun dapat mengurangkan masa dan kos, tidak dapat menghasilkan produk yang boleh memenuhi kebanyakan pengguna dari segi kualiti. Teknologi pengimbasan telah semakin matang dan kos perkakasan serta perisian pemvektor am semakin murah. Ini menjadikan pengimbasan satu kaedah alternatif yang boleh digunakan untuk penukaran data kadaster. Walau bagaimanapun, beberapa ciri yang spesifik kepada pelan kadaster tidak dapat diselesaikan oleh pemvektor am dan ini memerlukan pemprosesan lanjutan. Prototaip pemprosesan lanjut ini sedang dilaksanakan dan ujian-ujian awal menunjukkan hasil yang positif.

1. Pengenalan

Geoinformatik merupakan satu bidang penting daripada beberapa bidang teknologi maklumat. Geoinformatik boleh ditakrifkan sebagai teknologi maklumat khusus untuk data ruang (spatial data); iaitu data yang mana kedudukannya di permukaan bumi boleh diketahui.

GIS (Geographic Information Systems) ialah alat geoinformatik yang semakin popular masa kini. GIS merupakan sistem berkomputer yang digunakan untuk memperoleh, menyimpan, mengolah, menganalisis dan mempamirkan data dan maklumat ruang. GIS membolehkan maklumat penting untuk perancangan, pembangunan, penyenggaraan dan pengurusan diperolehi dengan pantas untuk pembuatan keputusan. Antara gunapakai GIS ialah pengurusan dan penilaian harta tanah; perancangan bandar dan wilayah; pengurusan sumber asli; pengurusan utiliti; penilaian kesan alam sekitar; pemodelan ekosistem; penganalisan pasaran; pengurusan dan penganalisan perniagaan; pengurusan pengangkutan dan telekomunikasi; dan ketenteraan.

Data memainkan peranan yang penting di dalam sesuatu perlaksanaan GIS. Kedapatan data yang berkualiti dan komperhensif membolehkan sistem membuat pertimbangan situasi dan mencadangkan tindakan yang lebih rasional dan efektif bagi

sesuatu masalah yang dihadapi. Walau bagaimanapun, mendapatkan data yang relevan, jitu dan mutakhir dalam konstrain belanjawan dan masa merupakan cabaran yang paling besar.

Perolehan data GIS dari salinan keras selalunya melalui proses pendigitan. Kaedah ini memang diakui murah dan senang dipelajari, tetapi sangat memakan masa dan terdedah kepada beberapa kesilapan terutama kesilapan pendigit yang telah keletihan. Kaedah alternatif melalui pengimbasan mempunyai beberapa kebaikan dan dengan turunya kos alat pengimbas, kaedah ini menjadi sangat kompetitif.

Kertas kerja ini membincangkan kaedah perolehan data lot melalui pengimbasan. Sumber data yang dipilih ialah salinan keras pelan akui. Di Bahagian Kedua, kita akan membincangkan masalah yang hendak diselesaikan. Bahagian Ketiga akan dimuatkan dengan tatacara penyelesaian masalah tersebut. Kita akan memberi satu contoh untuk memudahkan pemahaman tatacara penyelesaian, di Bahagian Keempat. Kertas kerja akan ditutup dengan beberapa rumusan dan cadangan di Bahagian Kelima.

2. Pernyataan Masalah

Pengimbasan lot kadaster dari pelan akui menghasilkan data raster. Walau bagaimanapun, kebanyakan gunapakai GIS memerlukan data kadaster berdigit dalam bentuk vektor. Dengan itu data raster tersebut perlu divektorkan menggunakan perisian pemvektor kegunaan am.

Masalah yang timbul ialah data vektor yang dihasilkan oleh pemvektor kegunaan am tidak memenuhi kehendak kebanyakan pengguna. Data tersebut masih rendah mutunya kerana terdapat vektor-vektor palsu (kecil), rerambut, segmen terputus dan sebagainya (Gambarajah 2.1). Kesalahan-kesalahan ini perlu diperbaiki sebelum data ini boleh digunakan.

3. Pemprosesan Data Vektor

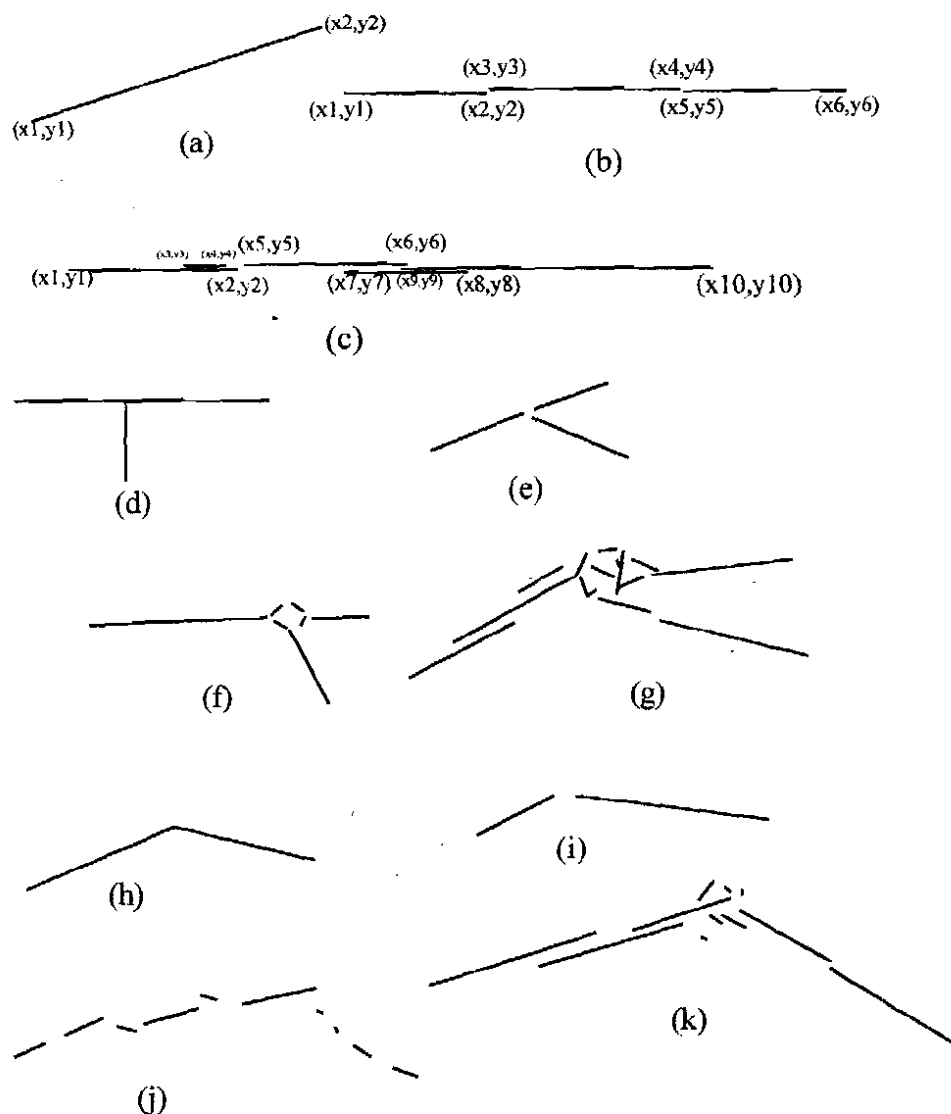
Langkah-langkah yang digunakan untuk memproses data-data vektor yang dihasilkan oleh pemvektor kegunaan am adalah seperti berikut.

A. Pemprosesan data segmen dengan menggunakan algoritma Peucker

Algoritma Peucker adalah algoritma yang digunakan untuk membuat generalisasi. Di sini, algoritma Peucker boleh digunakan untuk menentukan nod dan menghapuskan titik-titik palsu dan vektor palsu yang dihasilkan oleh pemvektor kegunaan am tidak diperlukan dalam segmen-segmen tertentu.

Dalam Gambarajah 3.1, untuk keadaan (a), titik-titik yang terletak di antara kedua-dua hujung segmen tersebut boleh diabaikan. Hanya kedua-dua hujung segmen perlu disambung dan membentuk satu vektor tunggal seperti yang ditunjukkan di (a'). Ini bermakna titik-titik persimpangan garisan tersebut boleh diabaikan. Bagi Gambarajah 3.1(b), vektor-vektor yang terletak di antara kedua-dua hujung segmen tersebut tidak boleh diabaikan. Titik-titik persimpangan adalah penting (atau dikenali sebagai nod) dan patut disimpan untuk mengekalkan bentuk objek seperti yang ditunjukkan dalam

(b'). Kesimpulan ini berasaskan kepada perwakilan grafik sempadan lot-lot kadaster di dalam pelan akui di mana jarak di antara nod lebih besar daripada satu nilai ambang.



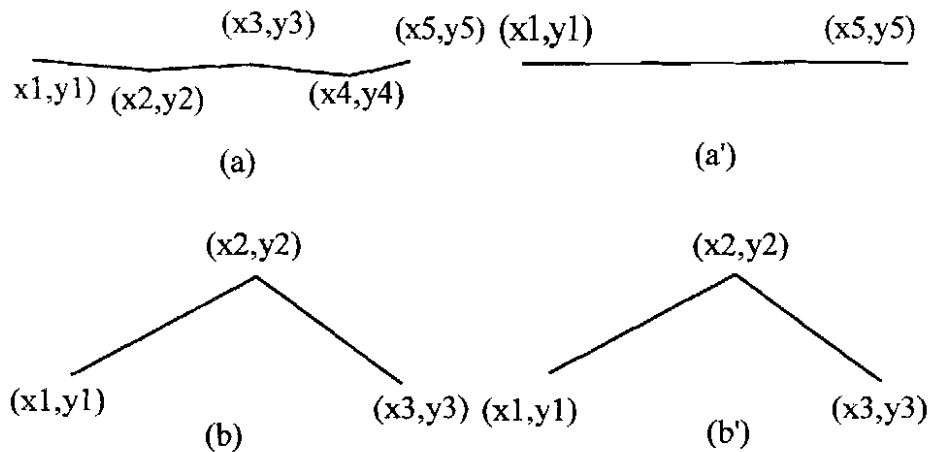
Gambarajah 2.1

Kebaikan penggunaan algoritma Peucker adalah seperti berikut.

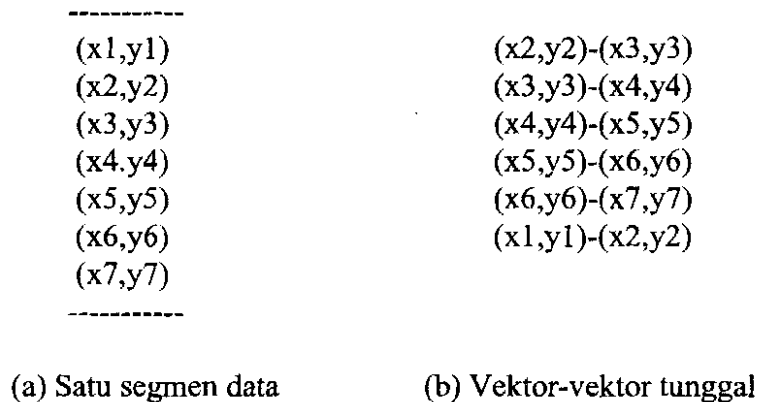
- la boleh digunakan untuk mengesan titik persimpangan mengikut tolerans yang telah ditetapkan oleh pengguna.
- Untuk peta kadaster, saiz simpanan data vektor boleh dikurangkan di antara 20 hingga ke 80 peratus.
- Bentuk peta dikekalkan.
- Boleh membuat penyesuaian garisan yang tidak lurus menjadi lurus semula.
- Boleh menghasilkan vektor panjang untuk pemprosesan yang seterusnya.
- Mengurangkan masa pemprosesan langkah-langkah seterusnya kerana saiz data vektor telah dikurangkan.

B. Penghapusan vektor palsu.

Selepas proses algoritma Peucker, data-data masih mengandungi vektor palsu. Langkah seterusnya ialah menghapuskan vektor palsu ini. Mula-mulanya, setiap segmen akan ditukarkan kepada vektor-vektor tunggal seperti yang ditunjukkan dalam Gambarajah 3.2.



Gambarajah 3.1



Gambarajah 3.2: Penjelmaan vektor segmen ke vektor tunggal.

Kemudian, magnitud vektor-vektor dikira menggunakan rumus.

$$d = [(x1-x2)^2 + (y1-y2)^2]^{1/2}$$

Akhirnya satu nilai tolerans, T telah ditetapkan dan semua vektor yang magnitudnya dibawah nilai T diabaikan. Hasilnya, semua vektor yang panjang akan dikekalkan.

Biasanya, vektor-vektor pendek yang telah dibuang adalah

- (1) vektor-vektor yang membentuk bulatan.
- (2) vektor-vektor yang membentuk huruf.
- (3) vektor-vektor yang terhasil disebabkan oleh rerambut.
- (4) vektor-vektor yang membentuk simbol.

C. Mengumpul titik-titik yang berdekatan dan mengesan nod-nod dalam peta tersebut.

Sekarang semua vektor yang disimpan adalah dalam bentuk vektor-vektor tunggal seperti Gambarajah 3.3.

$(x1,y1)-(x2,y2)$
 $(x3,y3)-(x4,y4)$
 $(x5,y5)-(x6,y6)$

Gambarajah 3.3: Susunan vektor-vektor dalam satu fail vektor

Adalah amat sukar untuk menentukan titik-titik yang berdekatan di antara satu sama lain menggunakan data dalam bentuk ini. Data tersebut, perlu disusun semula.

Mula-mula, bina satu fail koordinate dibina berpandu kepada fail vektor tersebut. Fail koordinate mempunyai 3 medan, iaitu, X-AXIS, Y-AXIS dan KEDUDUKAN. X-AXIS dan Y-AXIS adalah nilai koordinate manakala KEDUDUKAN adalah rujukan kepada kedudukan vektor dalam fail vektor. Jadi, fail koordinate adalah seperti Gambarajah 3.4.

(1) x1, y1, 1
(2) x2, y2, 1
(3) x3, y3, 2
(4) x4, y4, 2
(5) x5, y5, 3
(6) x6, y6, 3

Gambarajah 3.4: Susuan fail vektor dalam bentuk fail koordinate dan rujukan

Seterusnya, isihan dilakukan ke atas file koordinate tersebut. Titik-titik yang berdekatan akan dikumpul dan menjadi satu titik sepunya. Kaedahnya adalah seperti berikut. Katakan, satu siri koordinate yang telah diisi adalah seperti di Gambarajah 3.5.

x5,y5,3
x3,y3,2
x1,y1,1
x2,y2,1
x4,y4,2
x6,y6,3

Gambarajah 3.5: Fail titik yang telah diisi.

Katakan daripada fail titik, didapati bahawa $(x3,y3)$ yang terletak di lokasi 2 dan $(x2,y2)$ yang terletak di lokasi 1 adalah titik sepunya. Maka purata nilai-nilai x dan y dikira dengan menggunakan

$$\text{Min}(x)=(x3+x2)/2$$

$$\text{Min}(y)=(y_3+y_2)/2$$

Seterusnya, nilai $\text{Min}(x)$ dan $\text{Min}(y)$ akan digantikan dengan koordinat vektor-vektor di fail vektor. Jadi koordinat-koordinat di fail vektor yang baru adalah seperti Gambarajah 3.6.

$$\begin{aligned} &(x_1, y_1) - (\text{Min}(x), \text{Min}(y)) \\ &(\text{Min}(x), \text{Min}(y)) - (x_4, y_4) \\ &(x_5, y_5) - (x_6, y_6) \end{aligned}$$

Gambarajah 3.6: Fail vektor yang baru.

Beberapa keadaan akan berlaku selepas proses ini.

- Garisan-garisan yang telah putus disambung semula.
- Nod-nod dapat dibentuk.
- Titik-titik sepunya dapat dikesan.

Kelemahan proses ini adalah

- garisan bersepadu akan terhasil.
- garisan menjadi satu titik. Garisan tersebut adalah rerambut.

D. Membuang rerambut dan titik-titik yang dibentuk oleh vektor

Dengan menggunakan algoritma yang sama seperti langkah B, semua vektor-vektor yang pendek akan dibuang semula. Vektor-vektor yang dibuang adalah

- (1) titik-titik yang muncul apabila proses mencari titik sepunya di C. Titik-titik tersebut sebenarnya adalah vektor-vektor yang mempunyai magnitud sifar. Bentuk umum vektor tersebut adalah $(x_k, y_k) - (x_k, y_k)$.
- (2) vektor-vektor pendek yang dihasilkan semasa proses C.

E. Membuang garisan-garisan bersepadu

Selepas proses (D), data-data vektor bersih terhasil. Walau bagaimanapun, masih terdapat kemungkinan wujudnya garisan-garisan sepunya. Keadaan ini terjadi disebabkan oleh garisan-garisan yang dijelaskan dengan warna merah dalam peta kadaster yang asal. Selepas proses pengimbasan, garisan merah diimbis masuk ke dalam fail dan menjadi satu garisan tunggal yang berhampiran dengan garis asal. Oleh kerana titik-titik garisan tersebut adalah terlalu berdekatan dengan titik-titik garisan asal, maka kedua-dua garisan bercantum selepas proses (C). Jadi, garisan tersebut hanya perlu disimpan sekali sahaja.

Beberapa langkah yang perlu diambil adalah seperti berikut.

- (1) Semua garisan-garisan dalam fail perlu diisih. Oleh kerana terdapat 4 medan dalam satu rekod vektor, maka keempat-empat medan tersebut perlu dijadikan kunci isihan. Di sini, keempat-empat kunci adalah

Xaxis1, Yaxis1, Xaxis2, Yaxis2.

- (2) Selepas proses isihan, fail vektor yang diisi akan kelihatan seperti Gambarajah 3.7.

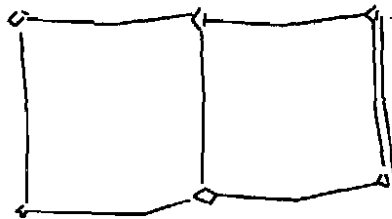
- (1)X1,Y1,X2,Y2
- (2)X1,Y1,X3,Y3
- (3)X2,Y2,X3,Y3
- (4)X2,Y2,X3,Y3
- (5)X3,Y3,X4,Y4

Gambarajah 3.7: Fail vektor.

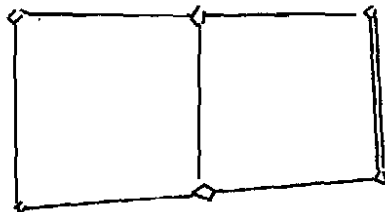
- (3) Oleh kerana rekod (3) dan rekod (4) adalah rekod yang sama, ia hanya perlu disimpan sekali sahaja. Proses ini dikesan mulai daripada rekod pertama hingga rekod yang terakhir dalam fail vektor tersebut. Selepas ini, semua garisan-garisan sepunya dibuang begitu sahaja.

4. Contoh

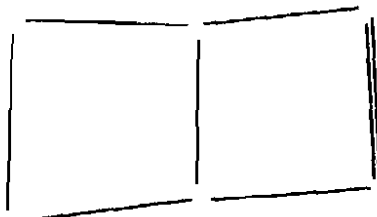
Satu sample data vektor untuk dua lot tanah sebelum pemprosesan



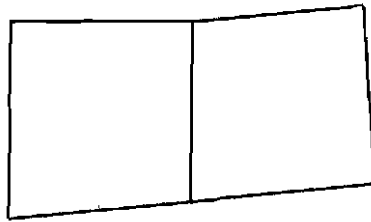
- (a) Selepas pemprosesan algoritma Peucker



- (b) Selepas pemecahan segmen dan penghapusan vektor-vektor palsu.



- (c) Kumpul titik berdekatan, sambung semula garisan-garisan yang berpecah dan buangkan semua garisan bersepadu.



5. Penutup

Kajian awal menunjukkan pegimbasan data pelan akui boleh dilakukan. Walau bagaimanapun, pemilihan pemvektor kegunaan am perlu dilakukan dengan berhati-hati. Penggunaan pemvektoran yang kurang baik menghasilkan vektor yang terlalu banyak mengandungi vektor palsu dan ini akan membebankan pemprosesan yang dibincangkan di dalam kertas kerja ini.

Penghargaan

Pengarang berterima kasih kepada Unit Penyelidikan dan Pembangunan UTM kerana membiayai sebahagian daripada penyelidikan ini melalui Vot No: 61558. Pengarang juga berterima kasih kepada Dr. Noordin Ahmad, di atas banyak pertolongan beliau.

Rujukan

Alameyko, S.; Beregov, B. and Kryuchkov, A. (1994) **Automatic map digitising: problems and solution**. Computing & Control Engineering Journal. IEEE. Vol: 5 p.33-9.

Boatto, L.; Consorti V.; Buono, M. D.; Zenzo, S. D.; Eramo V.; Esposito, A.; Melcarne, F.; Meucci, M.; Morelli, A.; Mosciatti M.; Scarci, S. and Tucci, M. (1992) **An interpretation system for land register maps**. Computer Vol: 25 p.25-33.

Dedic, A.; Murn, R. and Pecek, D. (1991) **Digitalization of large area drawings and maps**. 6th Mediterranean Electrotechnical Conference. Proceedings, 1991, Ljubljana, Slovenia.

Kasturi, R.; Fernandex, R.; Amlani, M.L. and Feng, W.C. (1989) . **Map data processing in geograpnic information systems**. Computer, vol:22, p10-21.

Lauterbach, B.; Ebi, N. and Besslich, P. (1992) **PROMAP- a system for analysis of topographic maps**. Proceedings IEEE Workshop on Applications of Computer Vision. IEEE Comput. Soc. Press. p46-55.

Mutsavi, M.T., Shirvaikar, M.V.; Ramanathan, E. and Nekovei, A.R. (1988) **Map processing methods: an automated alternative**. Proceedings of the Twentieth Southeastern Symposium on System Theory, USA.

Suzuki, S.. (1988). Graph-based vectorization method for line patterns.
Proceeding CVPR '88: The Computer Society Conference on Computer Vision and
Pattern Recognition, Ann Arbor, MI, USA